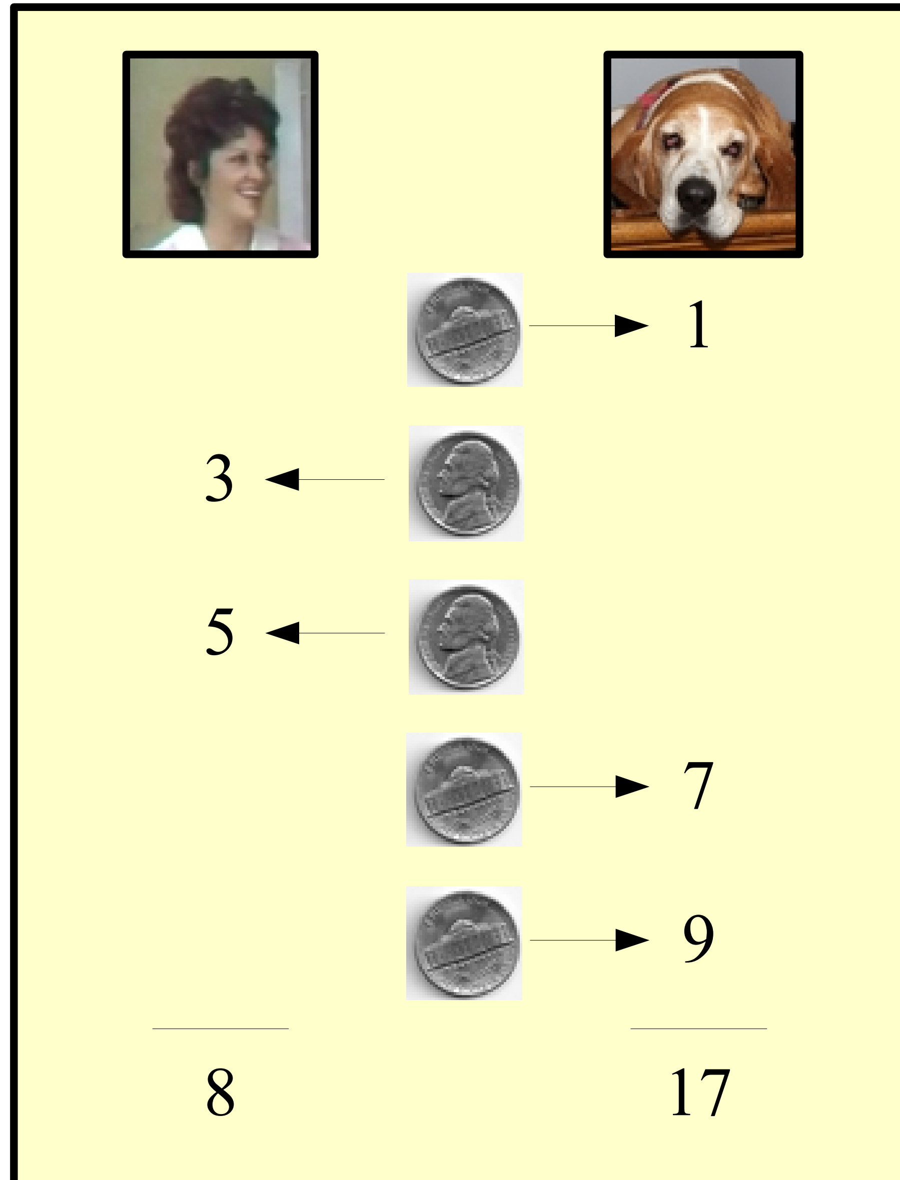
Topcoder SRM 639, D1, 250-Pointer "AliceGame"

James S. Plank
EECS Department
University of Tennessee

CS494/594 Class
August 28, 2017

The problem

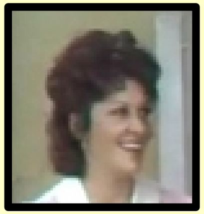
- Alice and Harvey are playing a coin-flip game.
- There are r rounds.
- Alice wins on heads, Harvey wins on tails.
- The first round is worth 1 point.
- Each subsequent round is worth 2 more points than the previous round.



The problem, continued

- You are given two numbers:
  - Alice's total score
  - Harvey's total score
- Return the minimum number of rounds that Alice could have won.
- Return -1 if the scores are impossible.

Example 0:



8

17

Answer is 2.
(Pictured on the last slide)
5 total rounds.
Alice wins rounds 2 and 3.

Prototype and Constraints

- **Class name**: `AliceGame`
- **Method**: `findMinimumValue()`
- **Parameters**:

a	`long long`	Alice's Total Score
b	`long long`	Harvey's Total Score

- **Return Value**: `long long`
- **Constraints**: a and b are between 0 and 10^{12}.
  - Which is 2^{40}, in case you've forgotten.

Observation #1

- *(a + b)* must be a perfect square. Why?

$$a + b = 1 + 3 + 5 + \ldots + 2r\text{-}1 = \sum_{i=1}^{r} (2i-1)$$

$$= \sum_{i=1}^{r} 2i - \sum_{i=1}^{r} 1$$

$$= \frac{2r(r+1)}{2} - r = r^2.$$

So our first test is to see if *(a+b)* is a square.

Observation #2

- Since a and b are limited by 10^{12}, and
- Since $r^2 = a+b$,
- Then: r is on the order of 10^6.

A solution that is linear in r will be fast enough.

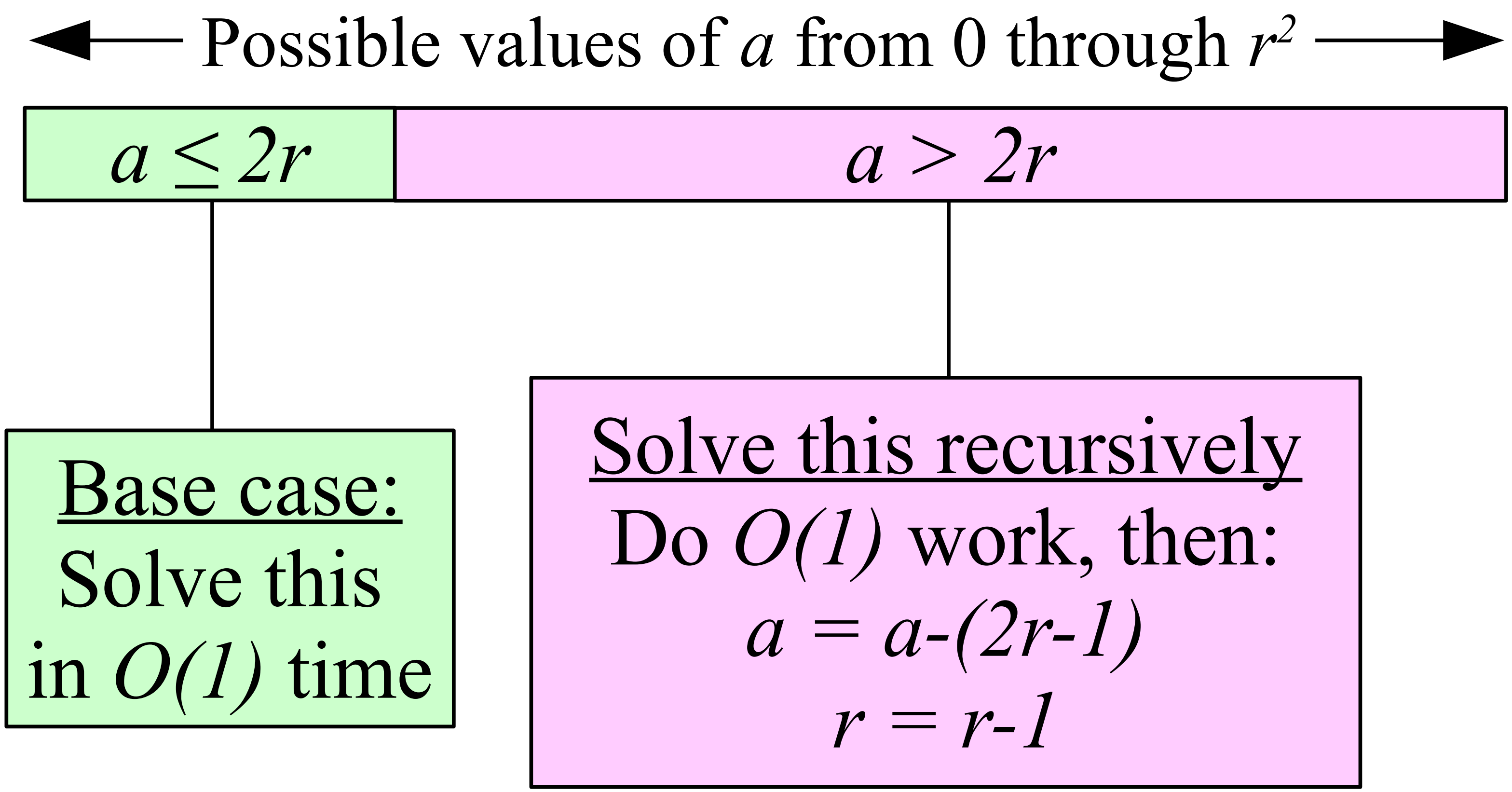
Observation #3

- Let r be the number of rounds.
  - $r^2 = (a+b)$
  - $a=2$ is unattainable
  - $a = r^2-2$ $(b=2)$ is unattainable
  - Everything else is attainable.

Example where $(a+b) = 16$
($r = 4$ rounds)

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	7
8	7+1
9	5+3+1
10	7+3
11	7+3+1
12	7+5
13	7+5+1
14	Impossible
15	7+5+3
16	7+5+3+1

Approach Using Recursion



This approach is $O(r)$.

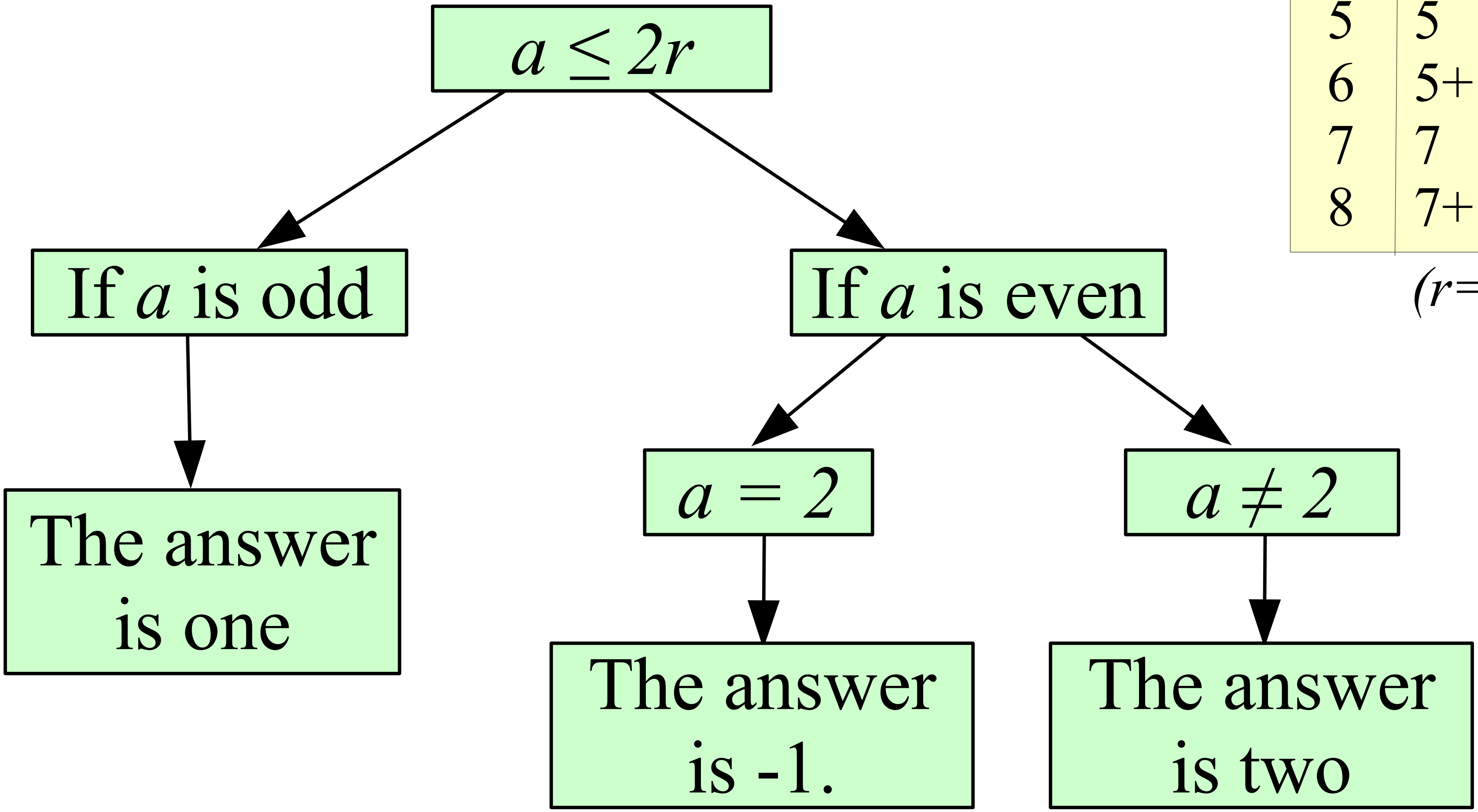
You'll note: if $a = r^2$, then $a-(2r-1) = (r-1)^2$

Base Case – Solving $a \leq 2r$

- Remember, the round scores are:
  - 1, 3, 5, …, 2r-1

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	7
8	7+1

(r=4)



How about $a > 2r$?

- Let's give round r to Alice, and then solve the problem recursively.
- Makes sense, because subtracting $(2r-1)$ will remove the most from Alice's score.

$$findMinimum(a, r) = 1 + findMinimum(a-(2r-1), r-1)$$

- Does it work?

How about $a > 2r$?

$$findMinimum(a, r) = 1 + findMinimum(a-(2r-1), r-1)$$

1	1
2	Impossible
3	3
4	3+1

$r=2$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	Impossible
8	5+3
9	5+3+1

$r=3$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	7
8	7+1
9	5+3+1
10	7+3
11	7+3+1
12	7+5
13	7+5+1
14	Impossible
15	7+5+3
16	7+5+3+1

$r=4$

How about $a > 2r$?

$$findMinimum(a, r) = 1 + findMinimum(a-(2r-1), r-1)$$

Works in all cases but this one!

1	1
2	Impossible
3	3
4	3+1

$r=2$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	Impossible
8	5+3
9	5+3+1

$r=3$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	7
8	7+1
9	5+3+1
10	7+3
11	7+3+1
12	7+5
13	7+5+1
14	Impossible
15	7+5+3
16	7+5+3+1

$r=4$

How about $a > 2r$?

$$findMinimum(a, r) = 1 + findMinimum(a-(2r-1), r-1)$$

Works in all cases but this one!

Where $a = 2r+1$.

$$(2r+1)-(2r-1) = 2$$

1	1
2	Impossible
3	3
4	3+1

$r=2$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	Impossible
8	5+3
9	5+3+1

$r=3$

1	1
2	Impossible
3	3
4	3+1
5	5
6	5+1
7	7
8	7+1
9	5+3+1
10	7+3
11	7+3+1
12	7+5
13	7+5+1
14	Impossible
15	7+5+3
16	7+5+3+1

$r=4$

So, let's fix that case

- When $a \leq 2r+1$, and $r > 3$, the answer is three:
  - Rounds 1, 2 and $r\text{-}1$.
  - Scores 1, 3 and $2r\text{-}3$
  - Whose sum is $2r+1$.

The Algorithm:

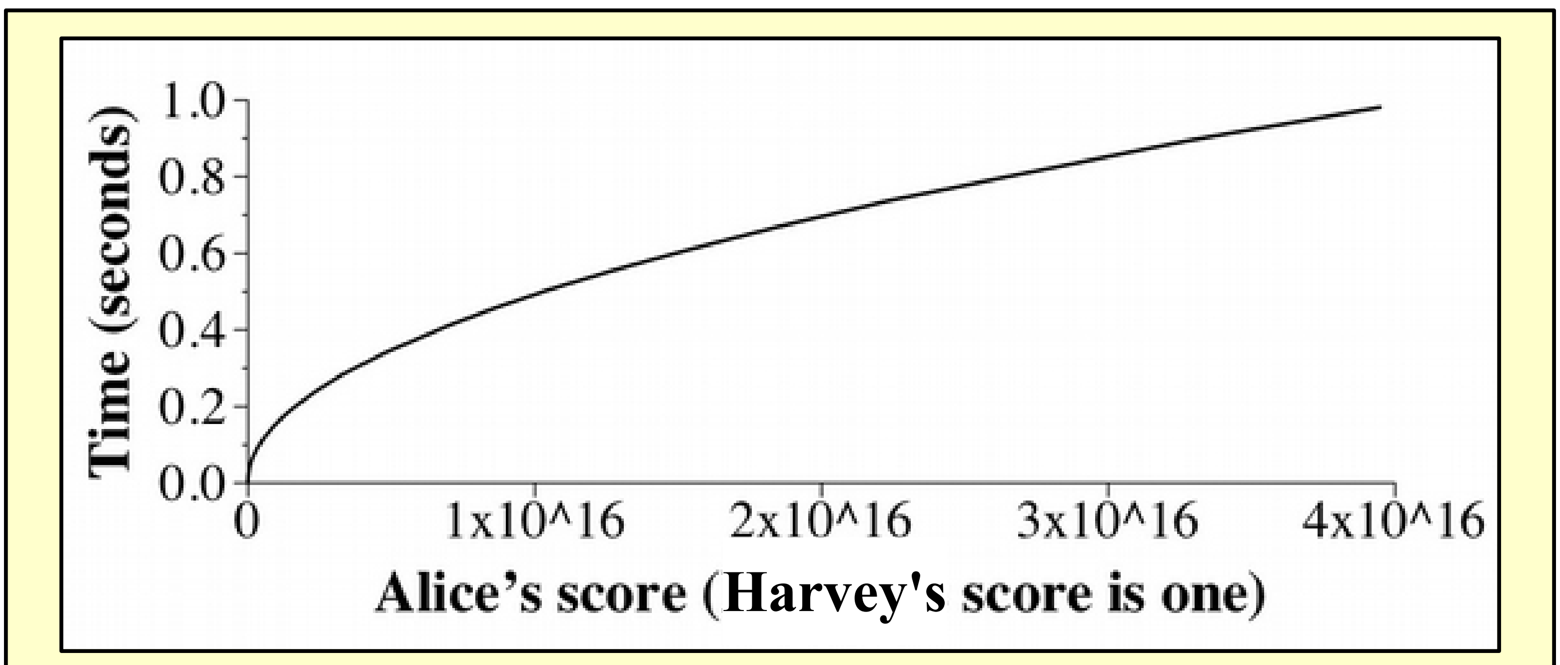
- If $(a+b)$ is not a perfect square, then return -1.
- Set r = ***sqrt***$(a+b)$.
- If $a = 2$ or $b = 2$, return -1.
- If $a = 0$, return 0.
- If $a < 2r$ and a is odd, return 1.
- If $a \leq 2r$ and a is even, return 2.
- If $a = 2r+1$, return 3.
- Otherwise, solve for $a = a-(2r-1)$ and $r = (r-1)$ and add one to the answer.

Base Cases

Recursion

Running Time:

- This iterates at most *r* times, so it is *O(r)*.
- Because $r \leq 10^6$, this runs fast enough to complete within Topcoder's limits.
- Recursion will fail, because nesting is *O(r)* too.
  - (Fails at *a* = 16,900,000,000)



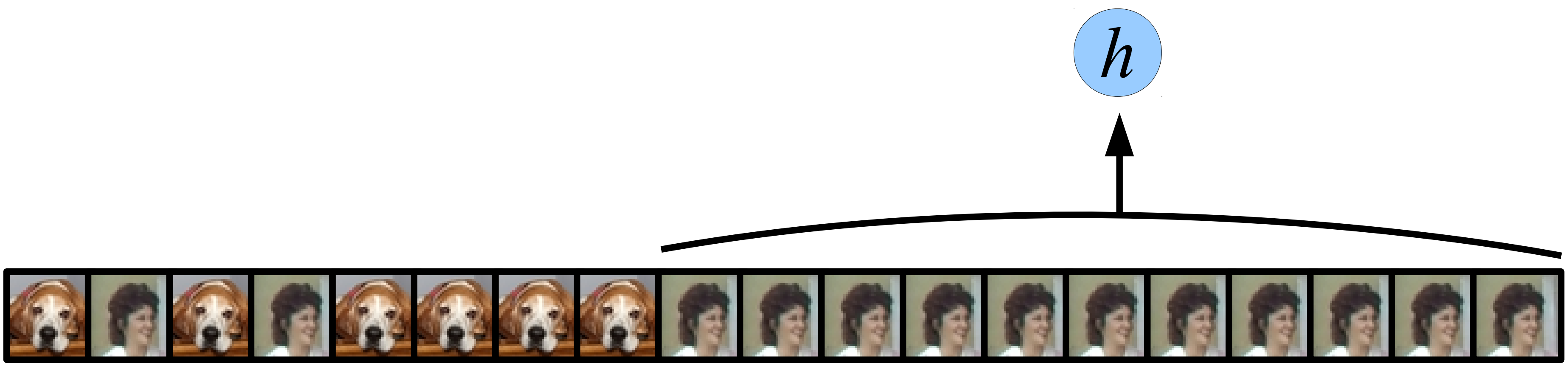
MacBook Pro
2.4 GHz
No optimization

Continuing in that vein:

- You can solve that algebraically if you want.
- However, when values get really large (think 2^{63}), can you rely on procedures like `sqrt()`?
- Think about it.

Making it faster:

- You can do this in *O(log(r))*.
  - Suppose the last *h* rounds go to Alice, but that the previous round goes to Harvey.
  - Then $r^2 - (r-h)^2 = (2rh - h^2)$ points go to Alice, and you can solve the remaining problem instantly.
  - Use binary search to find the largest legal value of *h*.



How did the Topcoders Do?

- This one was tricky:
  - 534 Topcoders opened the problem.
  - 496 (93%) submitted a solution.
  - 138 (28%) of the submissions were correct.
  - That's an overall percentage of 25.8%.
  - Best time was 4:22
  - Average correct time was 29:32.
- I suspect the *2r+1* part tripped people up.

Topcoder SRM 639, D1, 250-Pointer
"AliceGame"

James S. Plank
EECS Department
University of Tennessee

CS494 Class
August 28, 2018